System Safety for Highly Distributed Air Traffic Management

PI: Nancy Leveson, MIT

Co-PI: Chris Wilkinson, Honeywell

Problem Statement

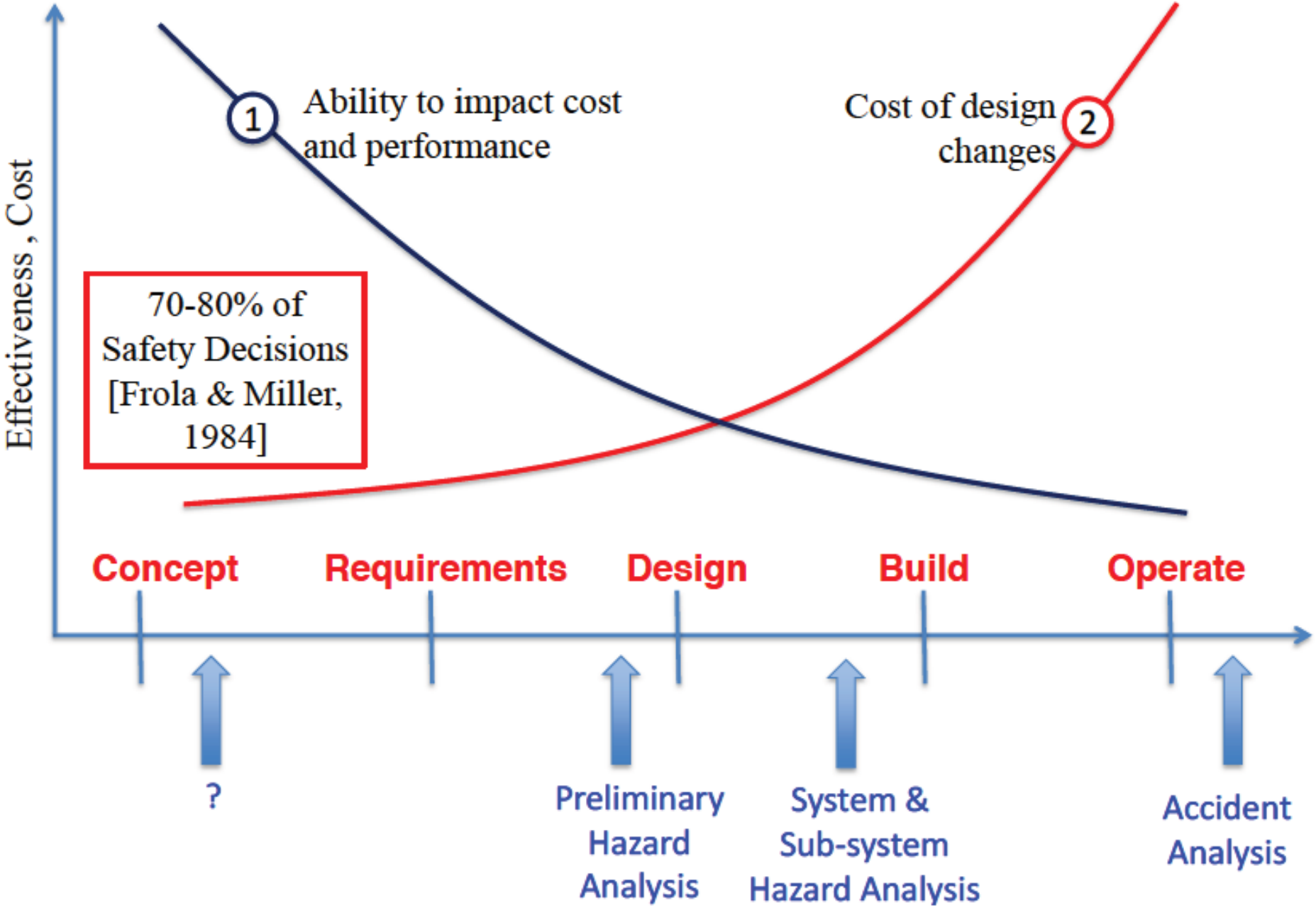
- Current flight-critical systems remarkably safe due to
  - Conservative adoption of new technologies
  - Careful introduction of automation to augment human capabilities
  - Reliance on experience and learning from the past
  - Extensive decoupling of system components
- Basically keep things simple and put up with inefficiencies

Problem Statement (2)

- NextGen introduces more complexity and potential for accidents:
  - Increased coupling and inter-connectivity among airborne, ground, and satellite systems
  - Control shifting from ground to aircraft and shared responsibilities
  - Use of new technologies with little prior experience in this environment
  - Increased reliance on software (allowing greater system complexity)
  - Human assuming more supervisory roles over automation, requiring more cognitively complex human decision making

Problem Statement (3)

- Attempts to re-engineer the NAS in the past have not been terribly successful and have been very slow, partly due to inability to assure safety.
- Question: What new methods for assuring safety will address challenges of NextGen that current methods do not?
- Hypotheses:
  - Rethinking how to engineer for safety is required to successfully introduce NextGen concepts
  - A new approach to safety based on systems theory can improve our ability to assure safety in these complex systems


Effectiveness , Cost
1
Ability to impact cost and performance
Cost of design changes
2
70-80% of Safety Decisions [Frola & Miller, 1984]
Concept
Requirements
Design
Build
Operate
?
Preliminary Hazard Analysis
System & Sub-system Hazard Analysis
Accident Analysis

Research Goals

- Create a hazard analysis method that works in concept development stage and supports safety-guided design to
  - Find flaws in NextGen concept documents (ConOps)
  - Evaluate the safety implications of alternative NextGen architectures.
  - Show how to derive verifiable system and software safety requirements from ConOps
  - Evaluate how the new approach would fit into the current FAA ATO Safety Management System
- Extend hazard analysis to include more sophisticated human factors
- Evaluate new analysis techniques by comparing results with the current state-of-the-art approach being used on NextGen

Traditional Ways to Cope with Complexity

1. Analytic Reduction
2. Statistics

Analytic Reduction

- Divide system into distinct parts for analysis

  Physical aspects → Separate physical components or functions

  Behavior → Events over time

- Examine parts separately and later combine analysis results

- Assumes such separation does not distort phenomenon
  - Each component or subsystem operates independently
  - Analysis results not distorted when consider components separately
  - Components act the same when examined singly as when playing their part in the whole
  - Events not subject to feedback loops and non-linear interactions

Human factors concentrates on the “screen out”

Engineering concentrates on the “screen in”

Not enough attention on integrated system as a whole

Analytic Reduction does not Handle

- Component interaction accidents
- Systemic factors (affecting all components and barriers)
- Software and software requirements errors
- Human behavior (in a non-superficial way)
- System design errors
- Indirect or non-linear interactions and complexity
- Migration of systems toward greater risk over time (e.g., in search for greater efficiency and productivity)

Standard Approach to Safety

- Reductionist
  - Divide system into components
  - Assume accidents are caused by component failure
  - Identify chains of directly related physical or logical component failures that can lead to a loss
  - Assume randomness in the failure events so can derive probabilities for a loss
- Forms the basis for most safety engineering and reliability engineering analysis:

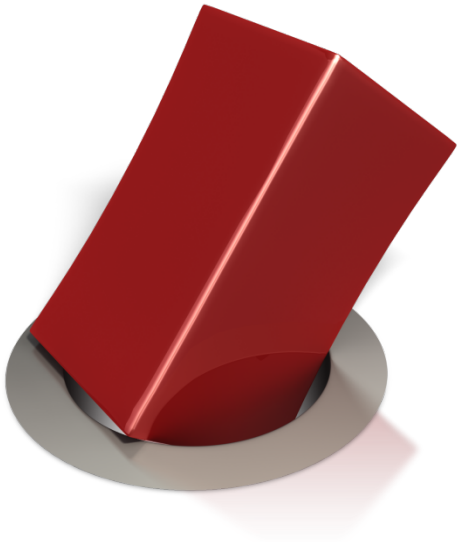
  FTA, PRA, FMEA/FMECA, Event Trees, etc.

  and design (concentrate on dealing with component failure):

  Redundancy and barriers (to prevent failure propagation),
  high component integrity and overdesign, fail-safe design, ….
- Note software does not fit: software does not “fail,” it simply does something that is unsafe in a particular context

Summary

- New levels of complexity, software, human factors do not fit into a reductionist, reliability-oriented world.
- Trying to shoehorn new technology and new levels of complexity into old methods will not work



- “But the world is too complex to look at the whole, we need analytic reduction”
- Right?

Systems Theory

- Developed for systems that are
  - Too complex for complete analysis
    - Separation into (interacting) subsystems distorts the results
    - The most important properties are emergent
  - Too organized for statistics
    - Too much underlying structure that distorts the statistics
    - New technology and designs have no historical information
- Developed for biology (von Bertalanffy) and engineering (Norbert Weiner)
- First used on ICBM systems of 1950s/1960s

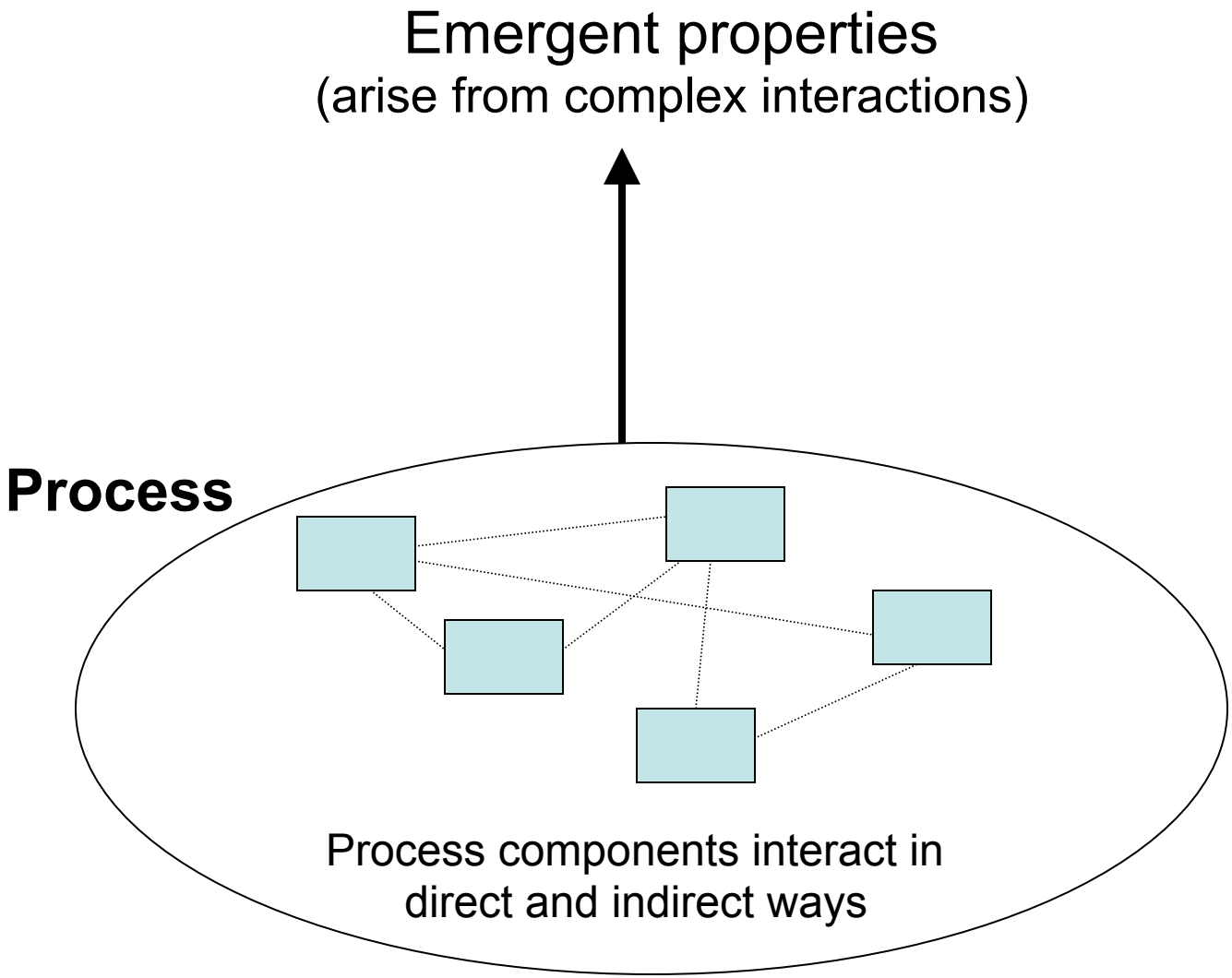
Systems Theory (2)

- Focuses on systems taken as a whole, not on parts taken separately
- Emergent properties
  - Some properties can only be treated adequately in their entirety, taking into account all social and technical aspects

    "The whole is greater than the sum of the parts"

  - These properties arise from relationships among the parts of the system

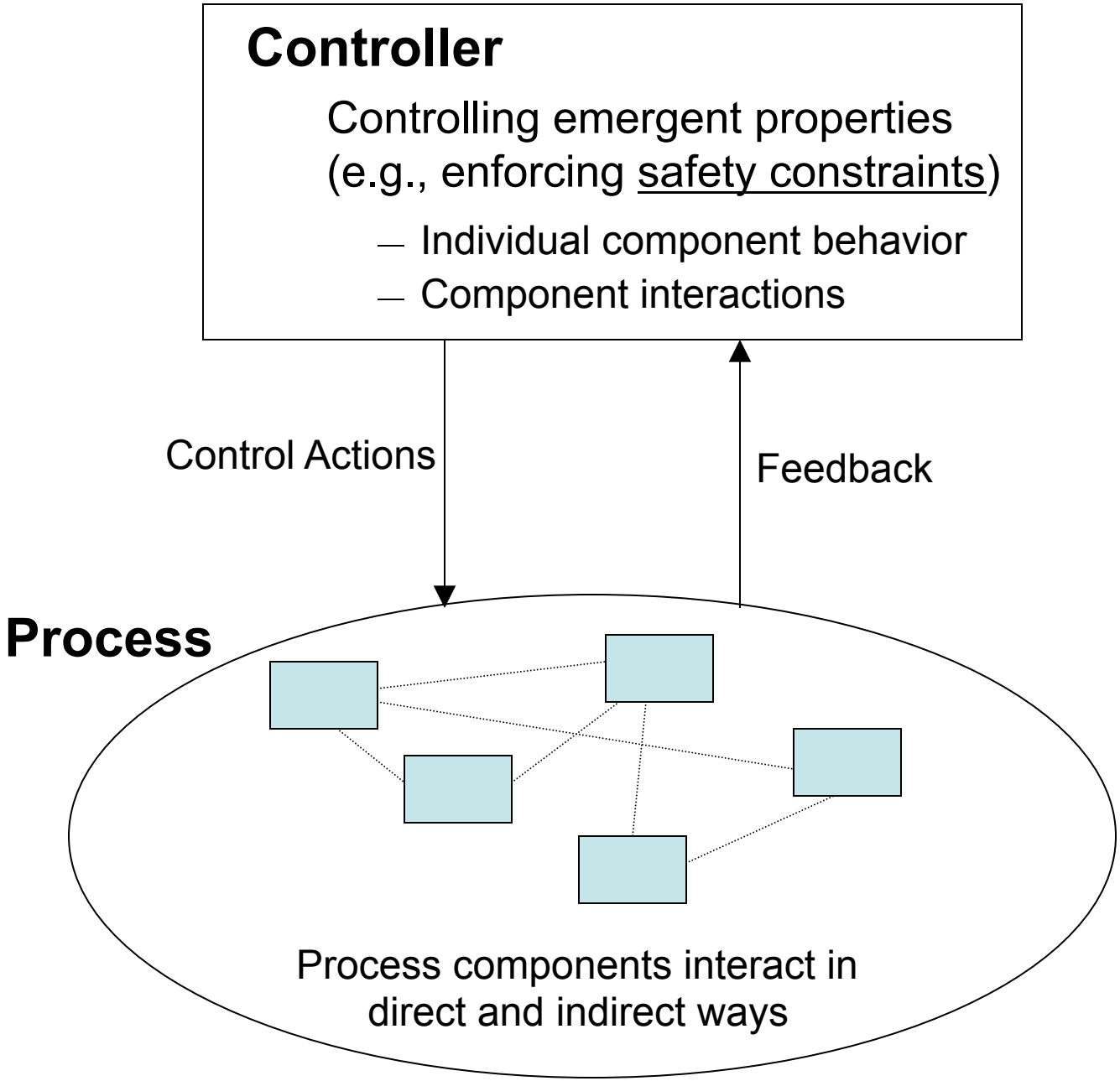
    How they interact and fit together



Safety is an emergent property

The STAMP Paradigm

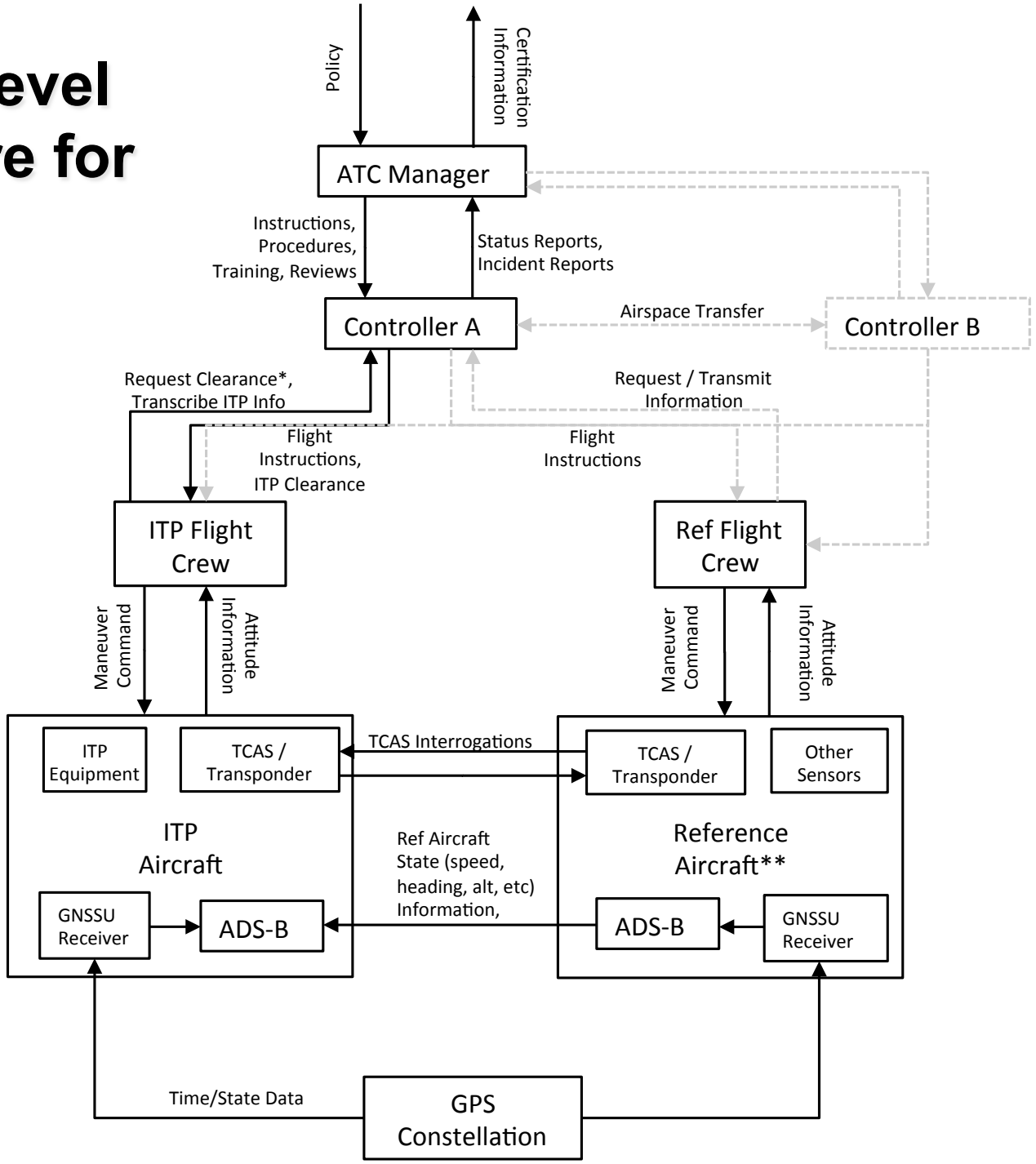
- Safety is a controllable system property if
  - Consider system at appropriate level
  - So can include **all** effects of system operations
  - Not just those attributable to component failure


Controller
Controlling emergent properties
(e.g., enforcing safety constraints)
– Individual component behavior
– Component interactions
Control Actions
Feedback
Process
Process components interact in
direct and indirect ways

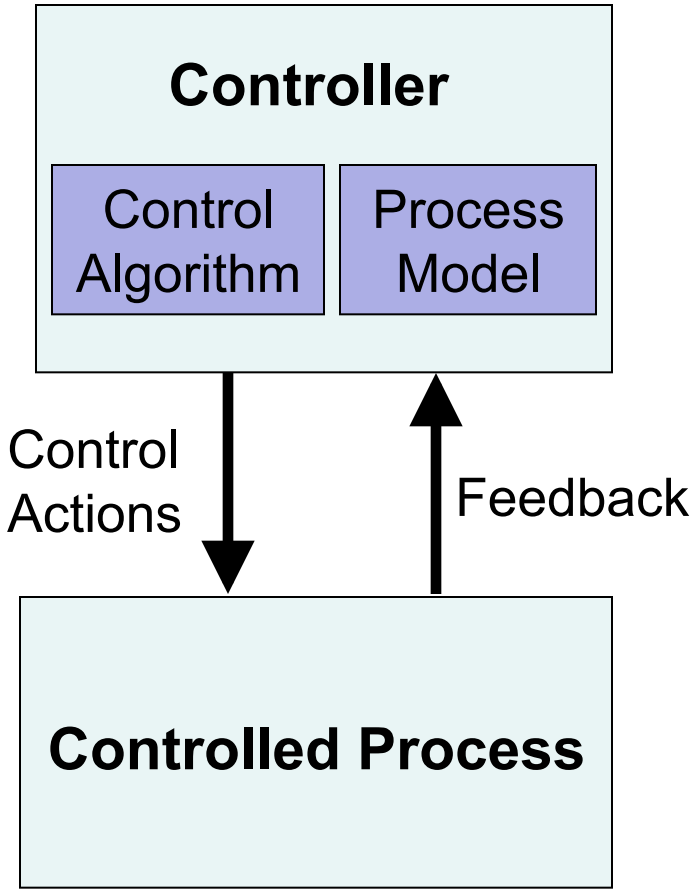
Controls/Controllers Enforce Safety Constraints

- Power must never be on when access door open
- Two aircraft must not violate minimum separation
- Aircraft must maintain sufficient lift to remain airborne
- Public health system must prevent exposure of public to contaminated water and food products
- Pressure in a deep water well must be controlled
- Truck drivers must not drive when sleep deprived

Example High-Level Control Structure for ITP



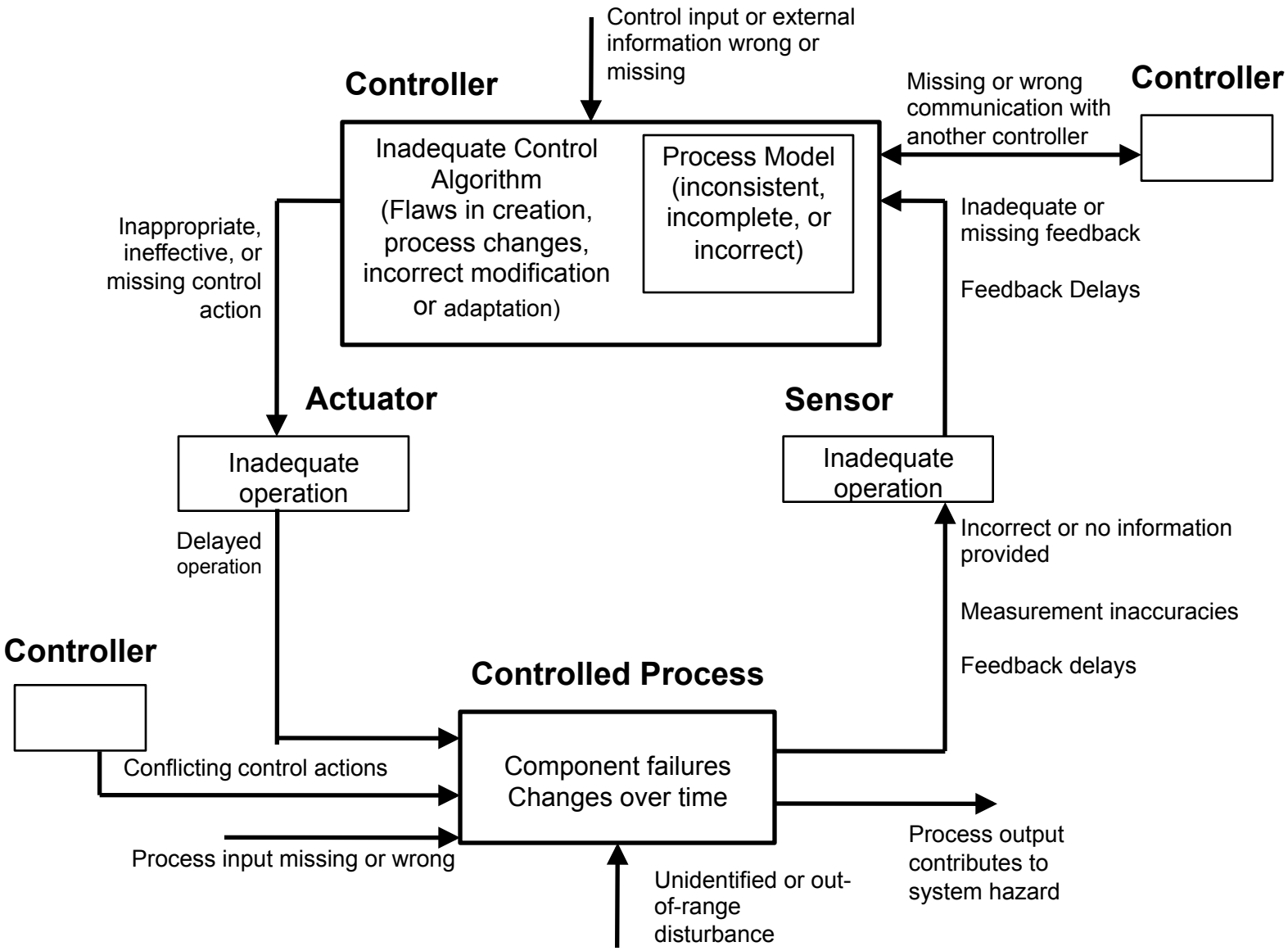
System Theoretic Process Analysis (STPA)



- Accidents often occur when process model inconsistent with state of controlled process (SA)
- Four types of unsafe control actions:
  - Control commands required for safety are not given
  - Unsafe ones are given
  - Potentially safe commands given too early, too late
  - Control stops too soon or applied too long
- Step 1: Identify unsafe control actions
- Step 2: Identify scenarios leading to unsafe control

(Leveson, 2003); (Leveson, 2011)

Identifying Causal Scenarios

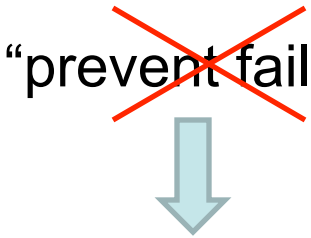


STAMP (System-Theoretic Accident Model and Processes)

- Defines safety as a control problem (vs. failure problem)
- Applies to very complex systems
- Includes software, humans, new technology
- Based on systems theory and systems engineering
- Expands the traditional model of the accident causation (cause of losses)
  - Not just a chain of directly related failure events
  - Losses are complex processes

Safety as a Dynamic Control Problem (STAMP)

- Events result from lack of enforcement of safety constraints in system design and operations
- Goal is to control the behavior of the components and systems as a whole to ensure safety constraints are enforced in the operating system
- A change in emphasis:



“enforce safety/security constraints on system behavior”

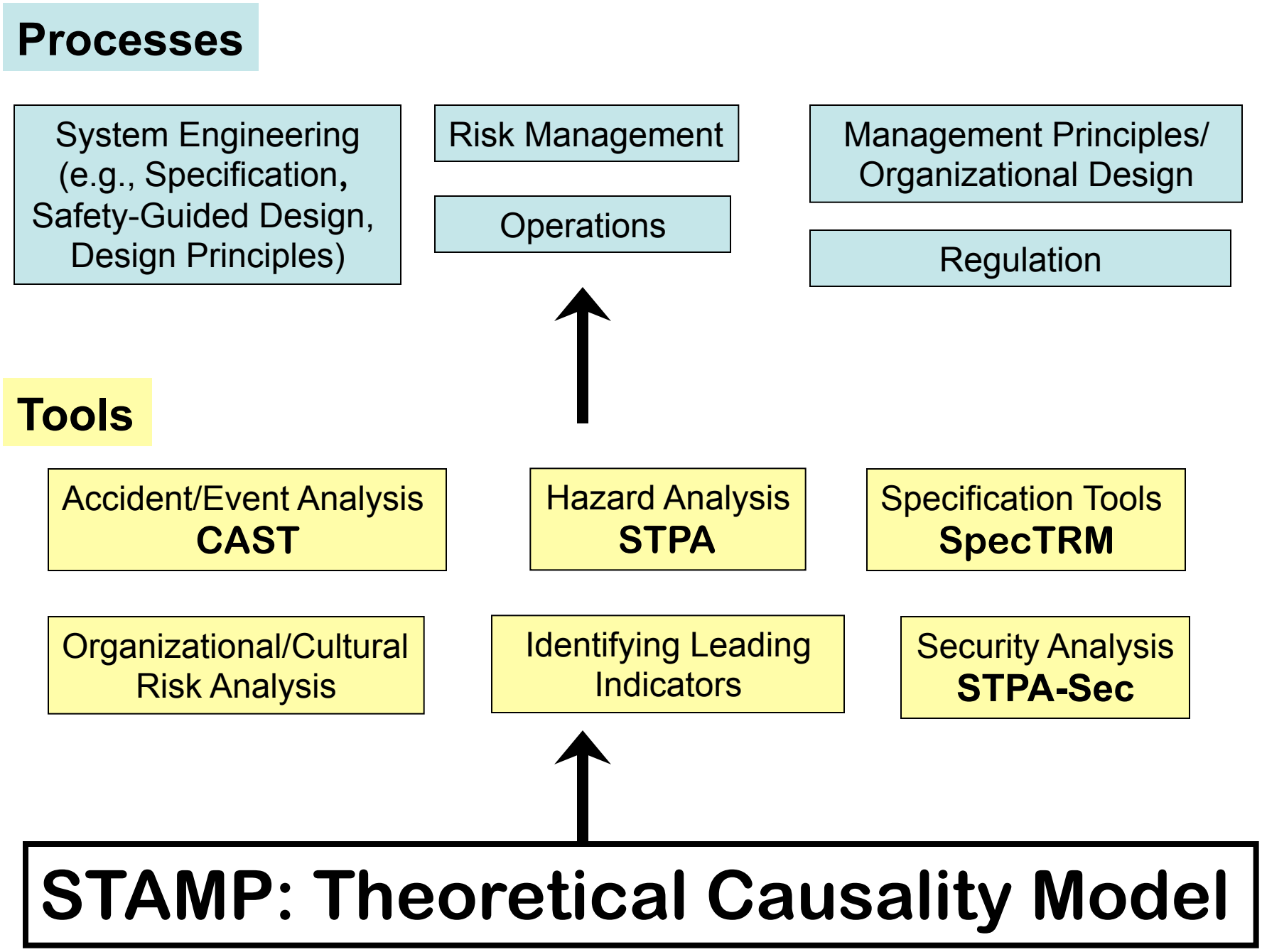
Changes to Analysis Goals

- Hazard analysis:
  - Ways that safety constraints might not be enforced

    (vs. chains of failure events leading to accident)

- Accident Analysis (investigation)
  - Why safety control structure was not adequate to prevent loss

    (vs. what failures led to loss and who responsible)


Processes
System Engineering (e.g., Specification, Safety-Guided Design, Design Principles)
Risk Management
Operations
Management Principles/ Organizational Design
Regulation
Tools
Accident/Event Analysis CAST
Hazard Analysis STPA
Specification Tools SpecTRM
Organizational/Cultural Risk Analysis
Identifying Leading Indicators
Security Analysis STPA-Sec
STAMP: Theoretical Causality Model

Is it Practical? Does it Work?

- STPA used in a large variety of industries around the world
- Most of these systems are very complex (e.g., the new U.S. missile defense system)
- In all cases where a comparison was made (to FTA, HAZOP, FMEA, ETA, etc.):
  - STPA found the same hazard causes as the old methods
  - Plus it found more causes than traditional methods
  - In some evaluations, found accidents that had occurred that other methods missed (e.g., EPRI)
  - Cost was orders of magnitude less than the traditional hazard analysis methods

LEARN 1 Grant (1) Results

1. Developed new analysis technique (based on STAMP and systems theory) to be used in early concept analysis
   - Rigorous procedure to construct the models from the ConOps
   - Analysis procedures to analyze the model
2. STECA (System-Theoretic Early Concept Analysis) uses ConOps to identify
   1. Missing, inconsistent, conflicting safety-related information
   2. Vulnerabilities, risks, tradeoffs
   3. Safety requirements for rest of system life cycle
   4. Potential design or architectural solutions for hazard scenarios
   5. Information needed by humans and by automation to operate safely (process models)

LEARN 1 Grant (2)

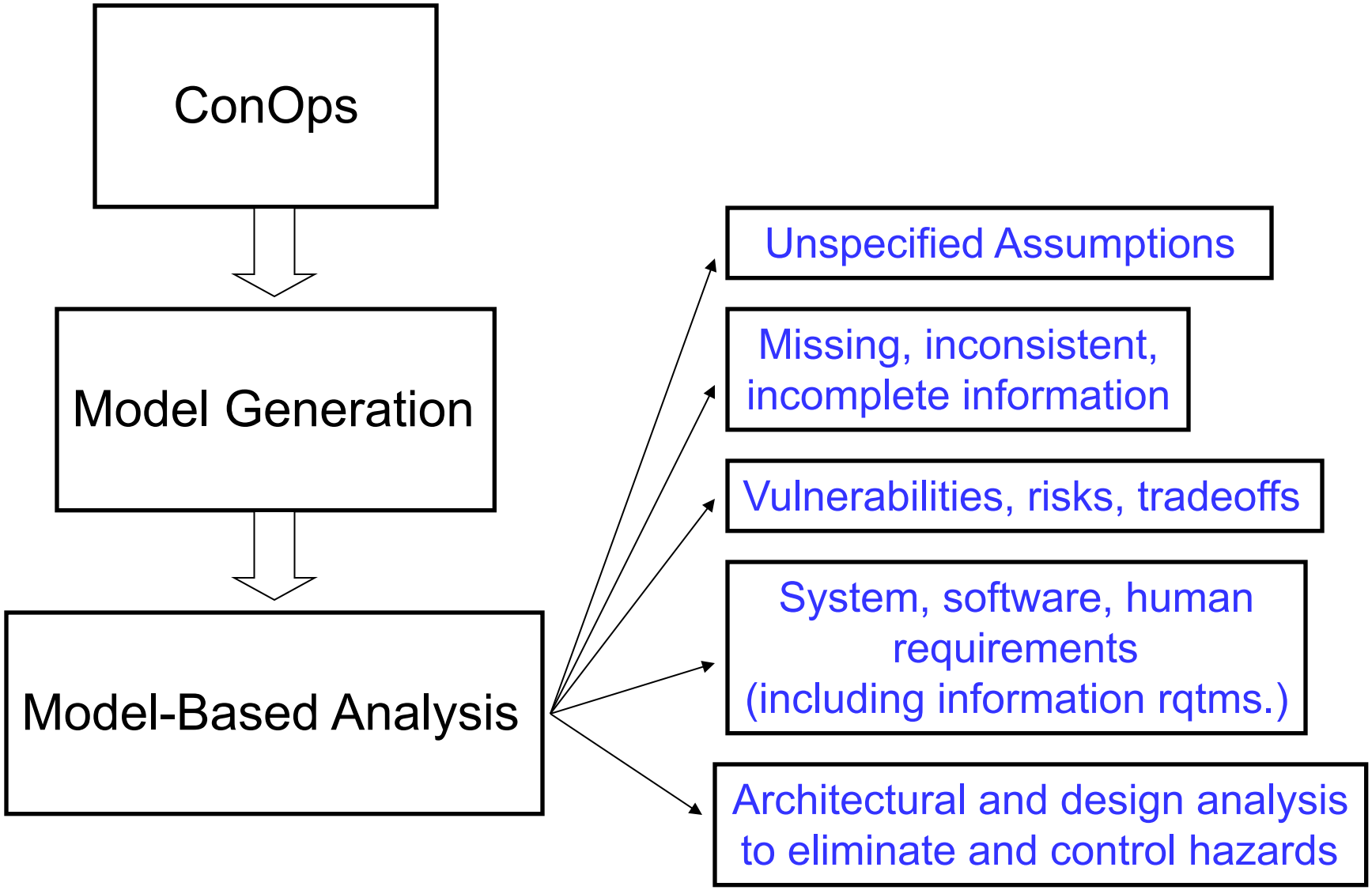
3. Demonstrated STECA on TBO (Trajectory-Based Operations) ConOps

4. Compared it to results of TBO PHA (Preliminary Hazard Analysis)

5. Extended STAMP hazard analysis to include some sophisticated human factors concepts (e.g., situation awareness)

Model-Based System Engineering

Dr. Cody Fleming



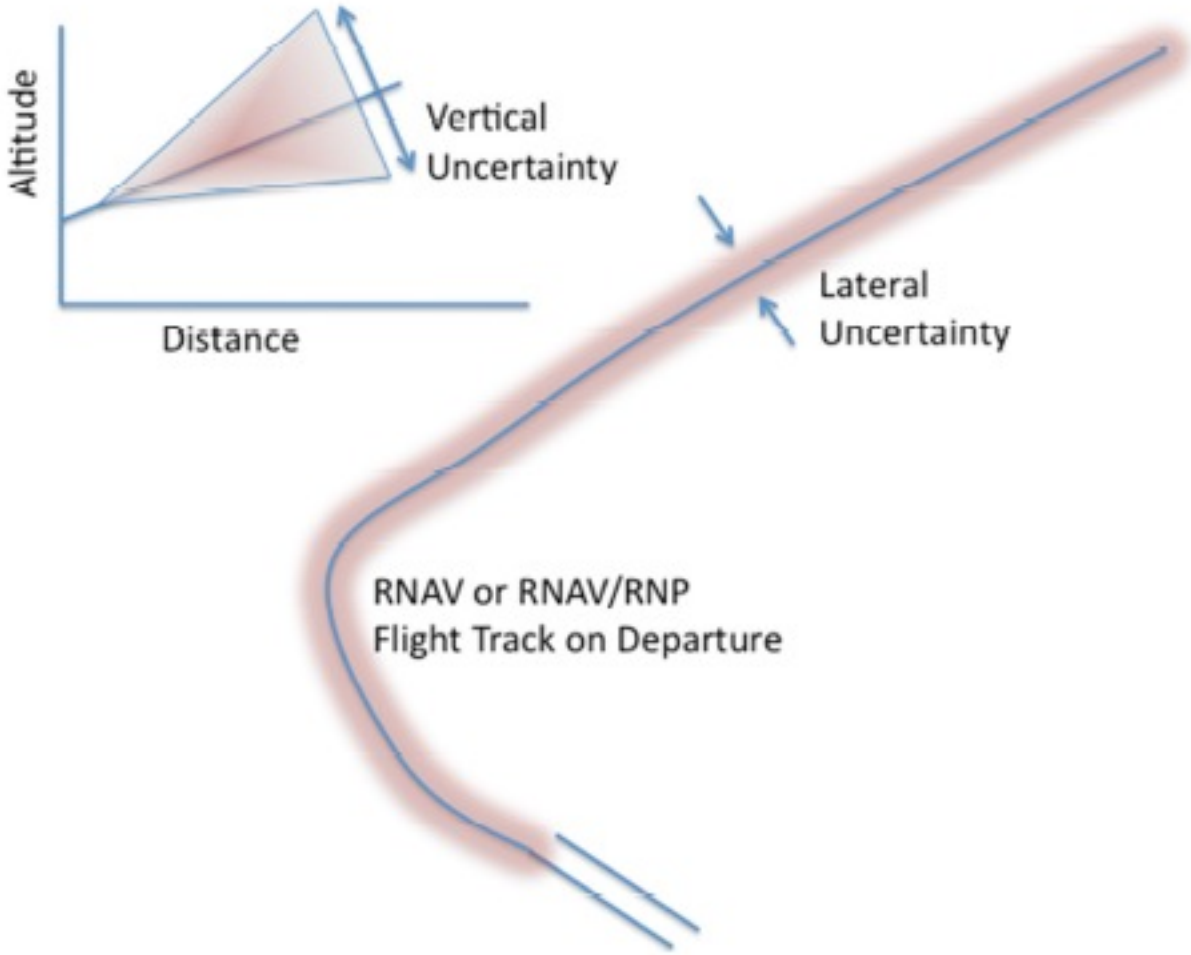
Joint Planning and Development Office

JPDO Trajectory-Based Operations (TBO) Study Team Report

December 4, 2011

Next Generation Air Transportation System
Joint Planning and Development Office

Application—TBO



[JPDO, 2011]

System Hazards

H1: Aircraft violate minimum separation (LOS or loss of separation, NMAC or near-midair collision)

H2: Aircraft enters uncontrolled state

H3: Aircraft performs controlled maneuver into ground

Safety Constraints

SC-1: Aircraft must remain at least TBD nautical miles apart en route [↑H-1]

SC-2: Aircraft position, velocity, must remain within airframe manufacturer defined flight envelope [↑H-2]

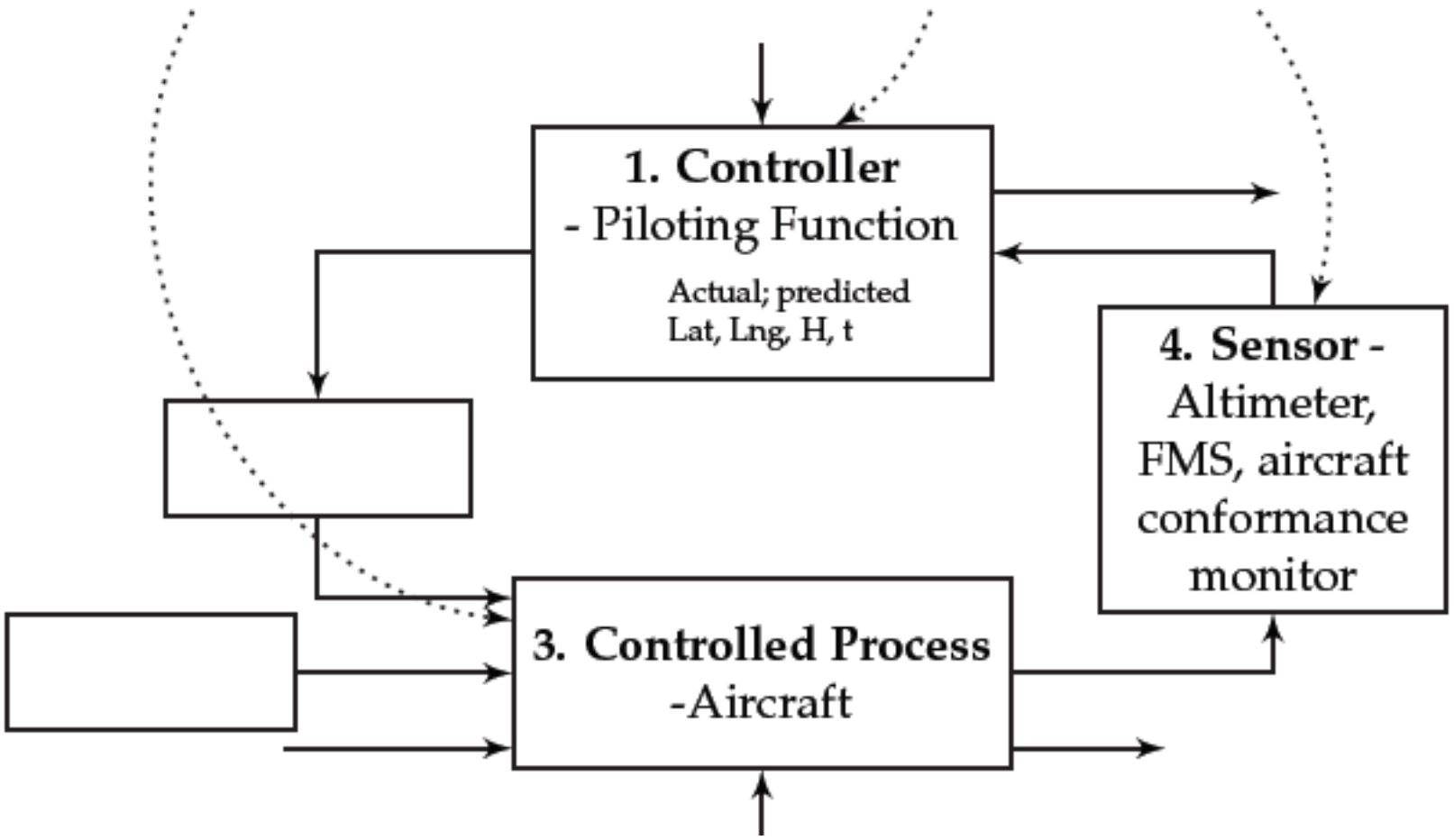
SC-3: Aircraft much maintain positive clearance with all terrain (this constraint does not include runways and taxiways) [↑H-3]

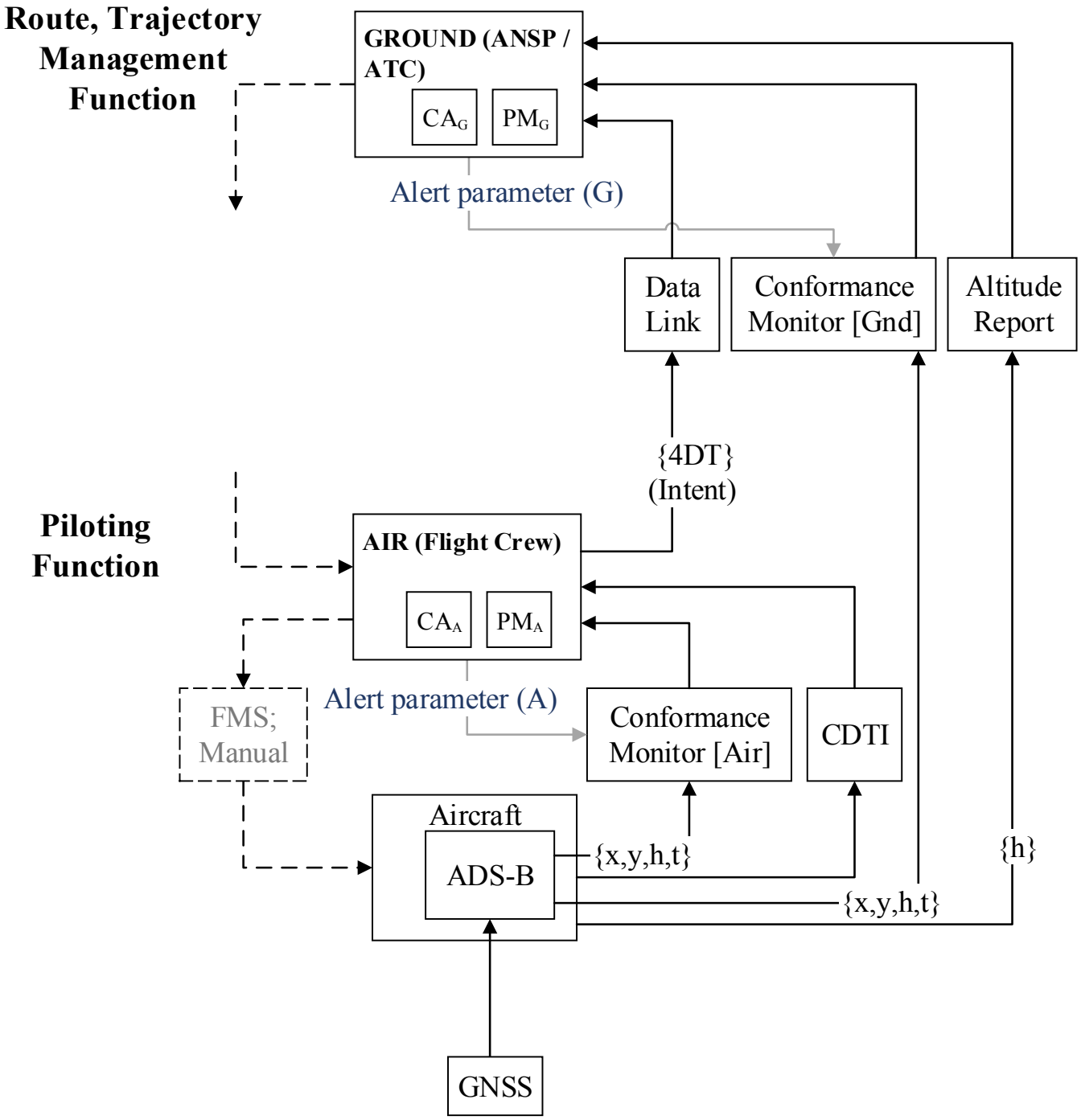
Identify Control Concepts

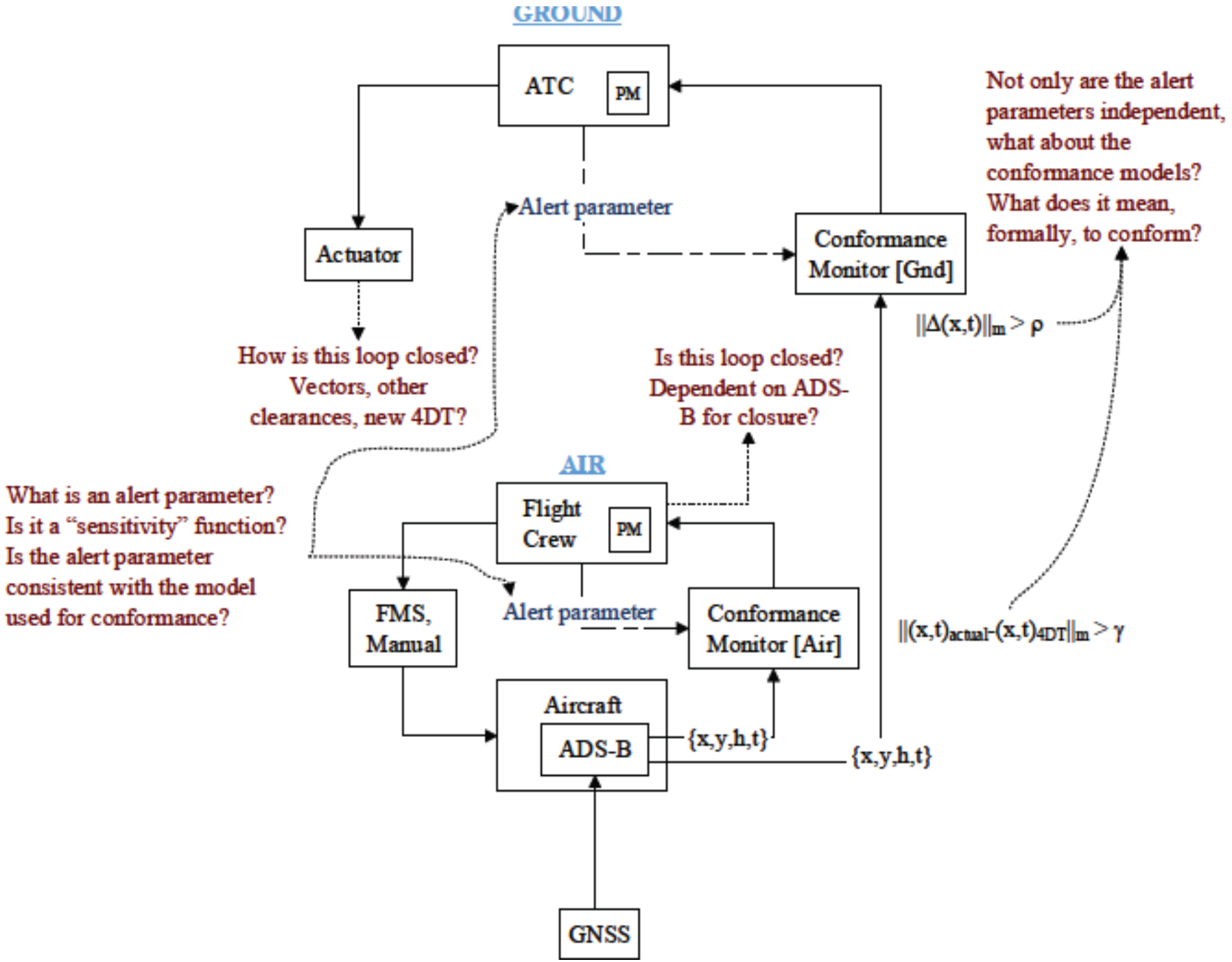
TBO conformance is monitored both in the aircraft and on the ground against the agreed-upon 4DT. In the air, this monitoring (and alerting) includes lateral deviations based on RNP..., longitudinal ..., vertical..., and time from the FMS or other "time to go" aids. [JPDO, 2011]

Subject	Conformance monitoring, Air automation
Role	Sensor
Behavior Type	Transmits binary or discretized state data to controller (i.e. measures behavior of process relative to thresholds; has algorithm built-in but no cntl authority)
	Sythesizes and integrates measurement data
Context	This is a decision support tool that contains algorithms to synthesize information and provide alerting based on some criteria.

TBO conformance is monitored both in the aircraft and on the ground against the agreed-upon 4DT. In the air, this monitoring (and alerting) includes lateral deviations based on RNP..., longitudinal ..., vertical..., and time from the FMS or other "time to go" aids. [JPDO, 2011]




Route, Trajectory Management Function
GROUND (ANSP / ATC)
CA_G
PM_G
Alert parameter (G)
Data Link
Conformance Monitor [Gnd]
Altitude Report
{4DT}
(Intent)
Piloting Function
AIR (Flight Crew)
CA_A
PM_A
FMS; Manual
Alert parameter (A)
Conformance Monitor [Air]
CDTI
Aircraft
ADS-B
{x,y,h,t}
{x,y,h,t}
{h}
GNSS


GROUND
ATC
PM
Actuator
Alert parameter
Conformance Monitor [Gnd]
Not only are the alert parameters independent, what about the conformance models? What does it mean, formally, to conform?
$\|\Delta(x,t)\|_m > \rho$
How is this loop closed? Vectors, other clearances, new 4DT?
Is this loop closed? Dependent on ADS-B for closure?
AIR
What is an alert parameter? Is it a "sensitivity" function? Is the alert parameter consistent with the model used for conformance?
Flight Crew
PM
FMS, Manual
Alert parameter
Conformance Monitor [Air]
$\|(x,t)_{actual}-(x,t)_{4DT}\|_m > \gamma$
Aircraft
ADS-B
{x,y,h,t}
{x,y,h,t}
GNSS

Analysis

1. Are the control loops complete?
2. Are the system-level safety responsibilities accounted for?
3. Do control agent responsibilities conflict with safety responsibilities?
4. Do multiple control agents have the same safety responsibility(ies)?
5. Do multiple control agents have or require process model(s) of the same process(es)?
6. Is a control agent responsible for multiple processes? If so, how are the process dynamics (de)coupled?

Analysis (2)

- Analysis properties defined formally, e.g.,
  - Gaps in responsibilities

$$(\forall \sigma_i \in \Sigma)\,(\exists \mathrm{c} \in \mathscr{C})\,[\mathrm{P}\,(\mathrm{c}, \sigma_i)]\,, \qquad (6)$$

  - Conflicts in responsibilities

$$(\forall H_i \in \mathcal{H})\,(\neg\exists \mathrm{c} \in \mathscr{C})\,[\mathrm{P}\,(\mathrm{c}, H_i) \wedge \mathrm{P}\,(\mathrm{c}, \mathcal{G})] \qquad (7)$$

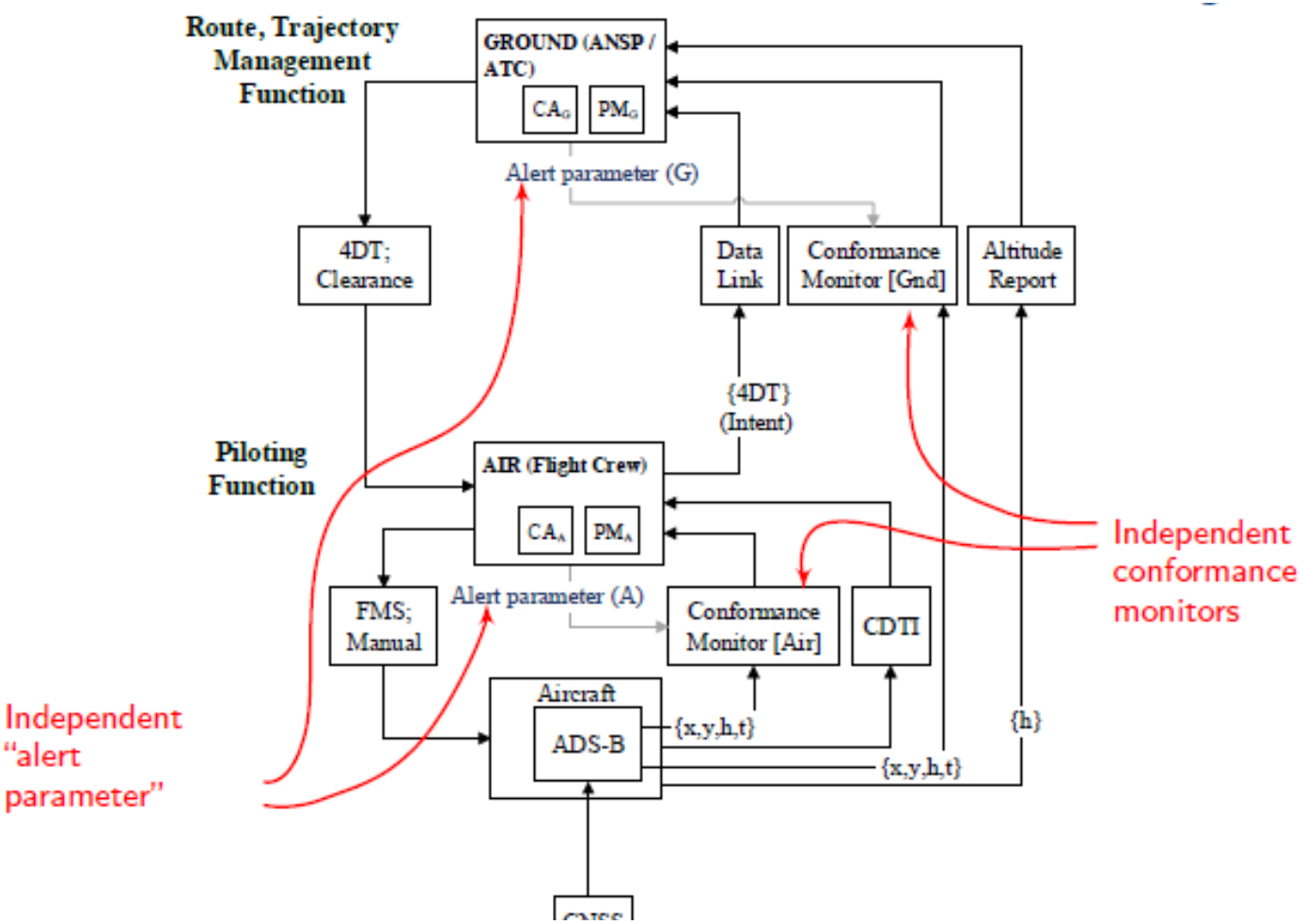
  - Coordination principle

$$(\forall \mathrm{c} \in \mathscr{C}_i)\,(\forall \mathrm{d} \in \mathscr{C}_j)\,\exists\,(\mathscr{P}\,(\mathrm{c}, \mathrm{d}) \vee \mathscr{P}\,(\mathrm{d}, \mathrm{c}))\,[\mathrm{A}\,(\mathrm{c}, \mathcal{V}_p) \wedge \mathrm{A}\,(\mathrm{d}, \mathcal{V}_p)]\,, \qquad (8)$$

  - Consistency principle

$$(\forall v \in \mathcal{V},\, \forall \mathrm{c} \in \mathscr{C}_i,\, \forall \mathrm{d} \in \mathscr{C}_j \mid \mathrm{A}\,(\mathrm{c}, v) \wedge \mathrm{A}\,(\mathrm{d}, v))$$
$$[\rho_i(a, v) \equiv \rho_j(a, v) \wedge G_i \equiv G_j] \qquad (9)$$

Coordination and Consistency



Software Requirements

Scenario 1:

The conformance monitoring model, i.e. the protected airspace volume, is insufficient or inadequate to maintain spacing

Causal Factors:

- This scenario might occur when the 4DT itself has a conflict;
- The conformance model is not updated to coincide with changing operations (e.g. en route vs. approach); [Model Condition, Observability Condition]
- The model does not ensure separation because additional traffic has joined the flow and constrained the airspace; [Model Condition, Observability Condition]
- Different aircraft have different conformance monitors

Software Requirements

Scenario 1:

The conformance monitoring model, i.e. the protected airspace volume, is insufficient or inadequate to maintain spacing

Requirements:

S1.1 4D Trajectories must remain conflict-free, to the extent possible

S1.2 Air traffic controllers, flight crews, and/or operations centers must be notified within TBD seconds of an overlap between any two 4D trajectories

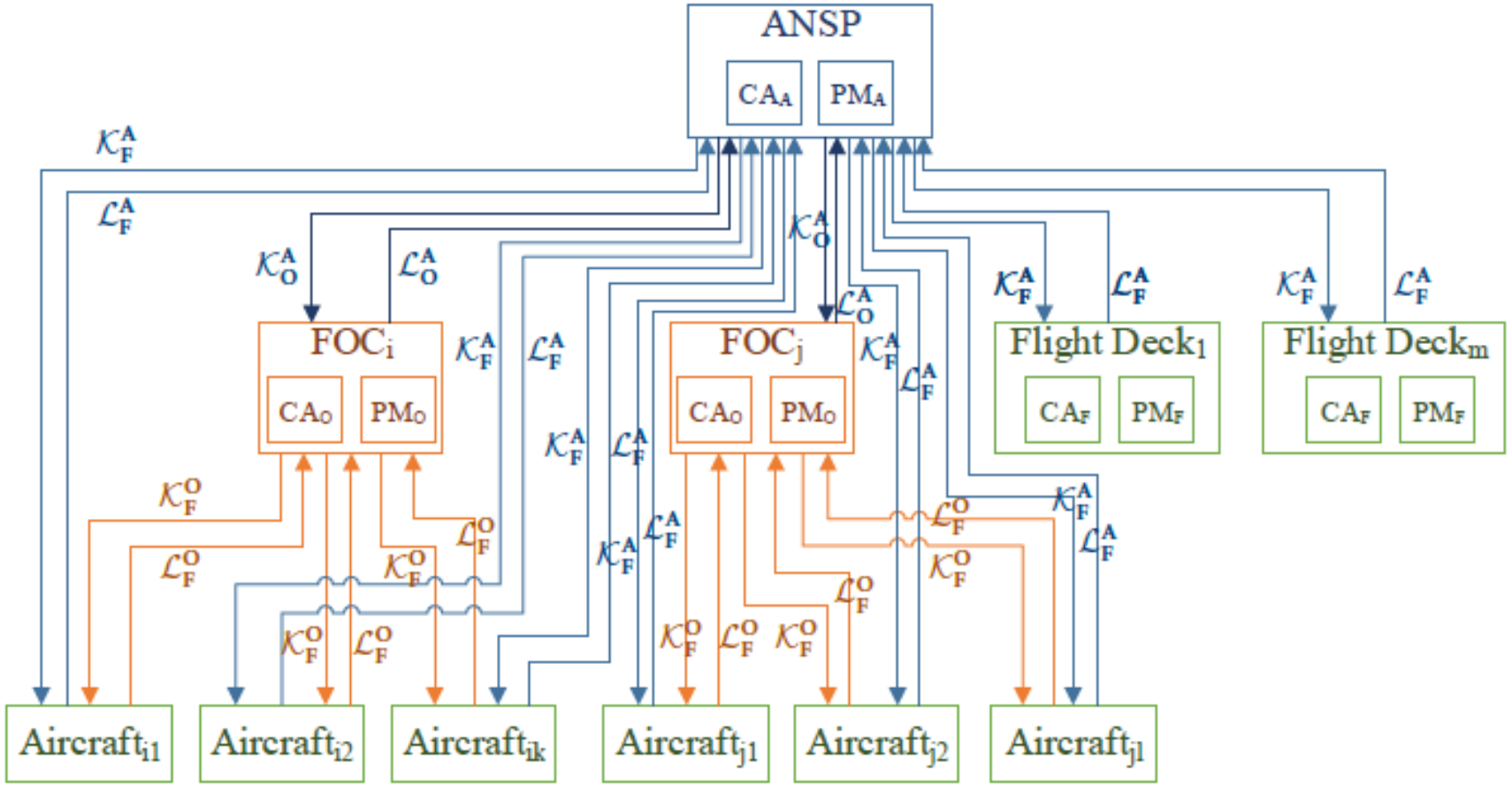
S1.3 Conformance volume must be updated within TBD seconds of change in separation minima

S1.4 Conformance monitoring software must be provided with separation minima information

S1.5 ...

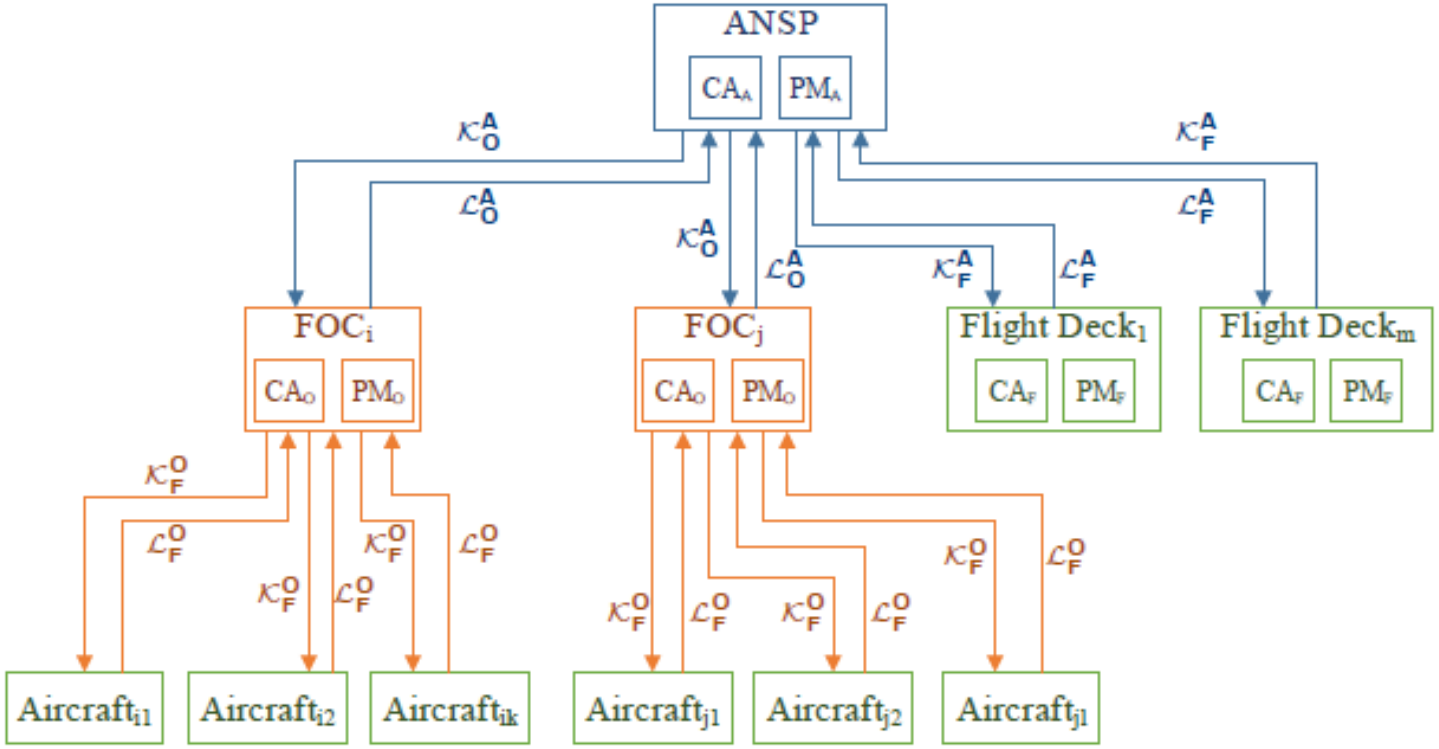
In same way specify requirements for hardware, human operators (pilots, air traffic controllers), interactions, etc.

Comparing Potential Architectures

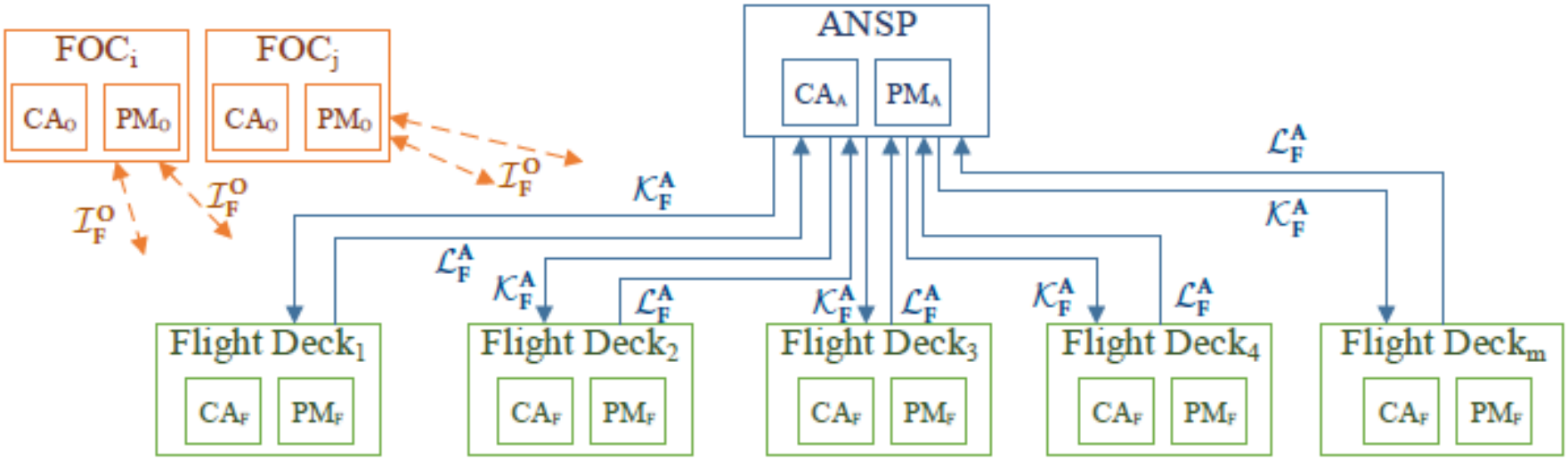


Control Model for Trajectory Negotiation

TBO Negotiation



Additional Requirement: $\mathcal{K}_F^A$ and $\mathcal{K}_F^O$ shall *not* occur simultaneously.



Alternative Control Model for Trajectory Negotiation

(Can compare architectures with respect to hazardous scenarios added or eliminated)

Recent PHA on TBO ConOps

Hazard Name	Hazard Desc.	Causes	Sev.	Like.	Assumed Mitigations	Mit. Str.	Risk	Justification
ADS-B Ground System Comm Failure	GBA does not receive ADS-B message	Receiver failure	H	L	Redundant equipment; certification requirements; etc.	M	M	Strength of mitigations depends on type of backup
GBA fails to recognize dynamic situation and is unable to find a solution	Software lacks robustness in its implement-ation that leads to inability to find a solution	Design flaw, coding error, insufficient software testing, software OS problems			Comprehensive system testing before cert. and operational approval. Pilot or controller could recognize in some cases.			Anything that is complex can lead to this situation

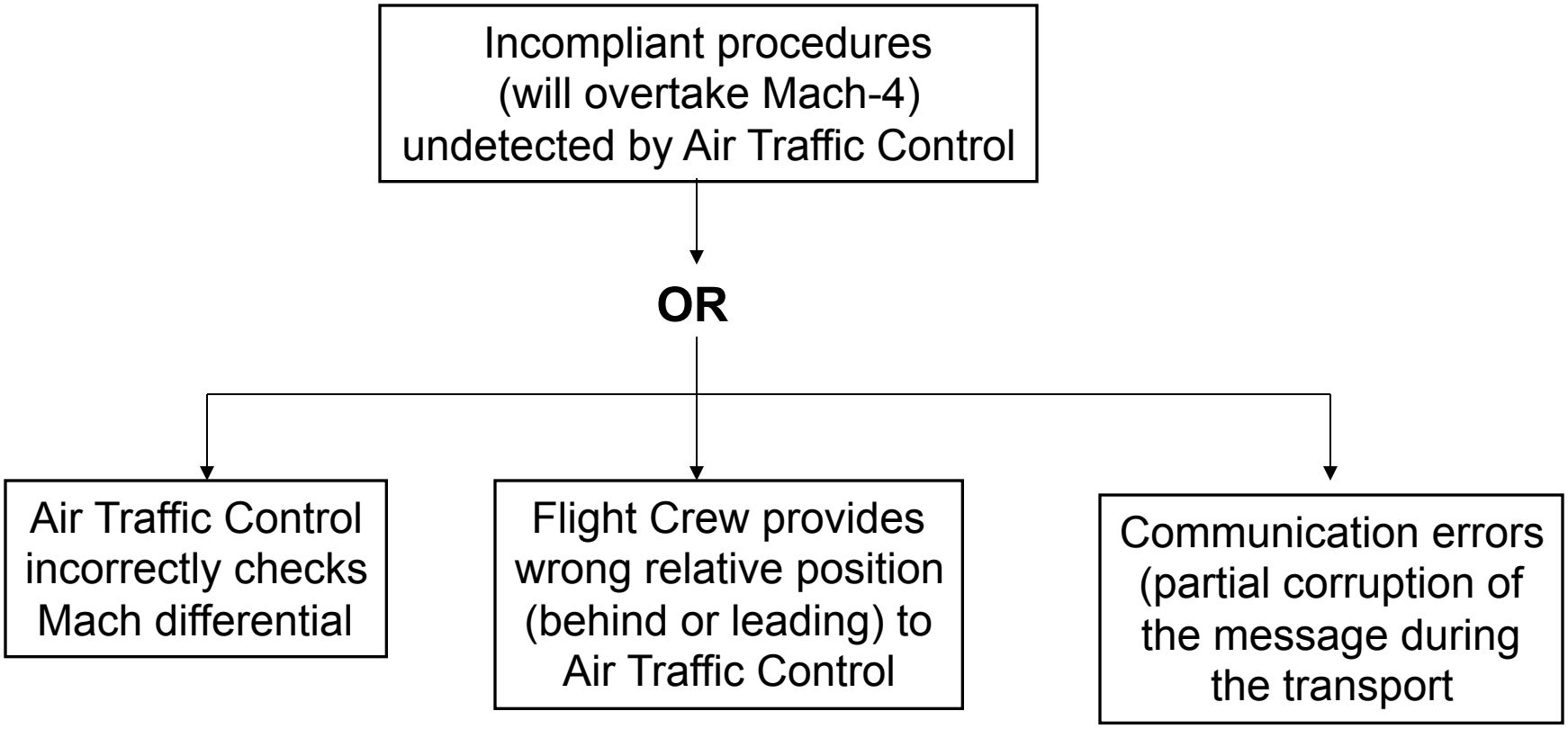
Comparison of STECA with Standard PHA

- PHA
  - Vague statements that do not help with designing safety into the system
  - Concentrates on component failure
- STECA:
  - Generates specific behavioral requirements for system, software, and humans to prevent hazards
  - Identifies specific scenarios leading to a hazard, even when do not involve a component failure
  - Provides means for analyzing potential designs and architectures and generating mitigations

Including Human-Controller in Hazard Analysis

- Cameron Thornberry (MIT Master's thesis)
- Leveraged principles from Ecological Psychology and basic cognitive models
- Two basic causal categories:
  - *Flawed detection and interpretation of feedback*
  - *Inappropriate affordance of action*
- Demonstrated on a proposed airspace maneuver called In-Trail Procedure that had been analyzed using STPA
  - Identified additional causal factors and unsafe control actions compared to RTCA analysis
  - Same ideas used in our TBO analysis

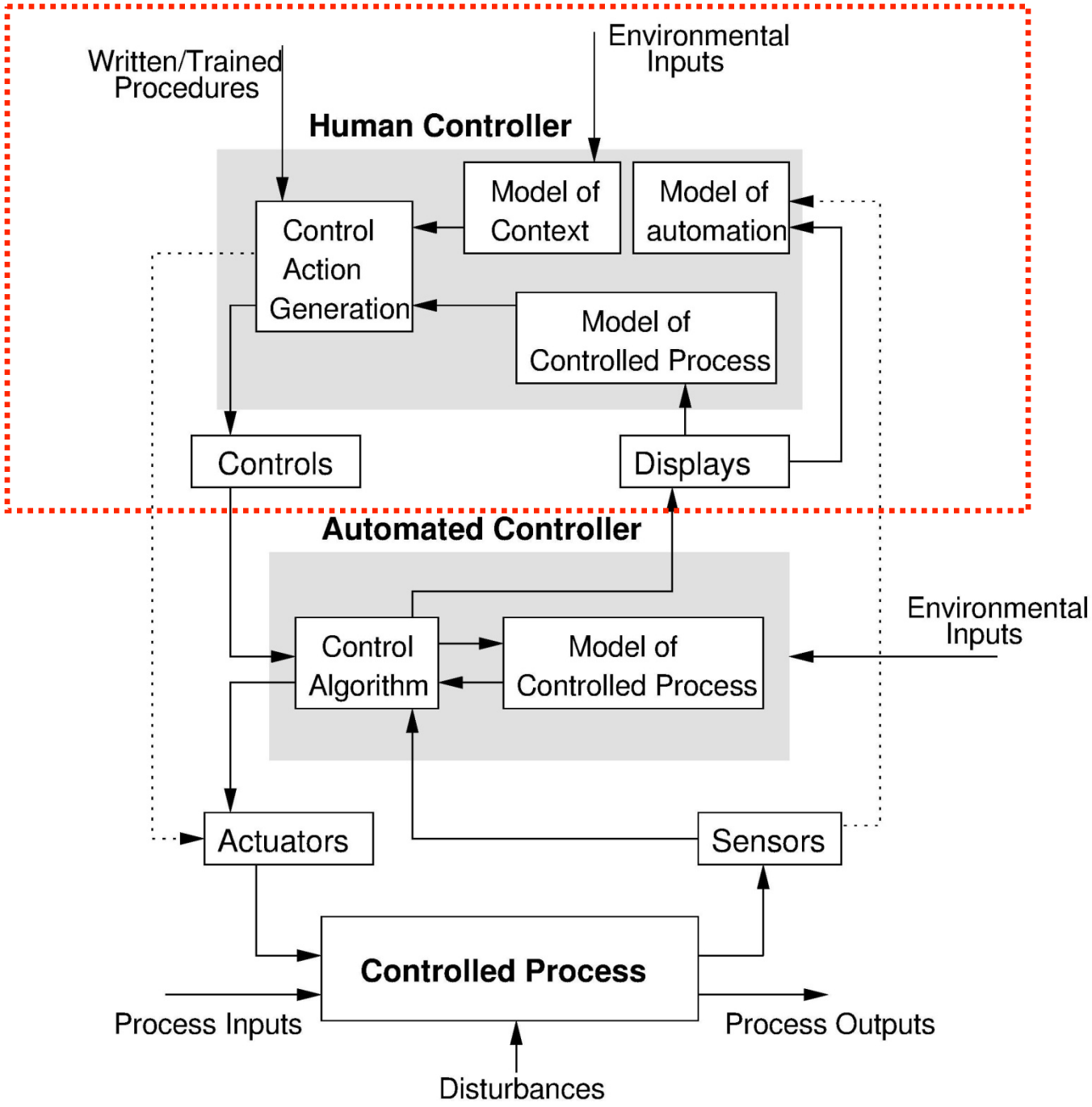
Human Factors in Hazard Analysis

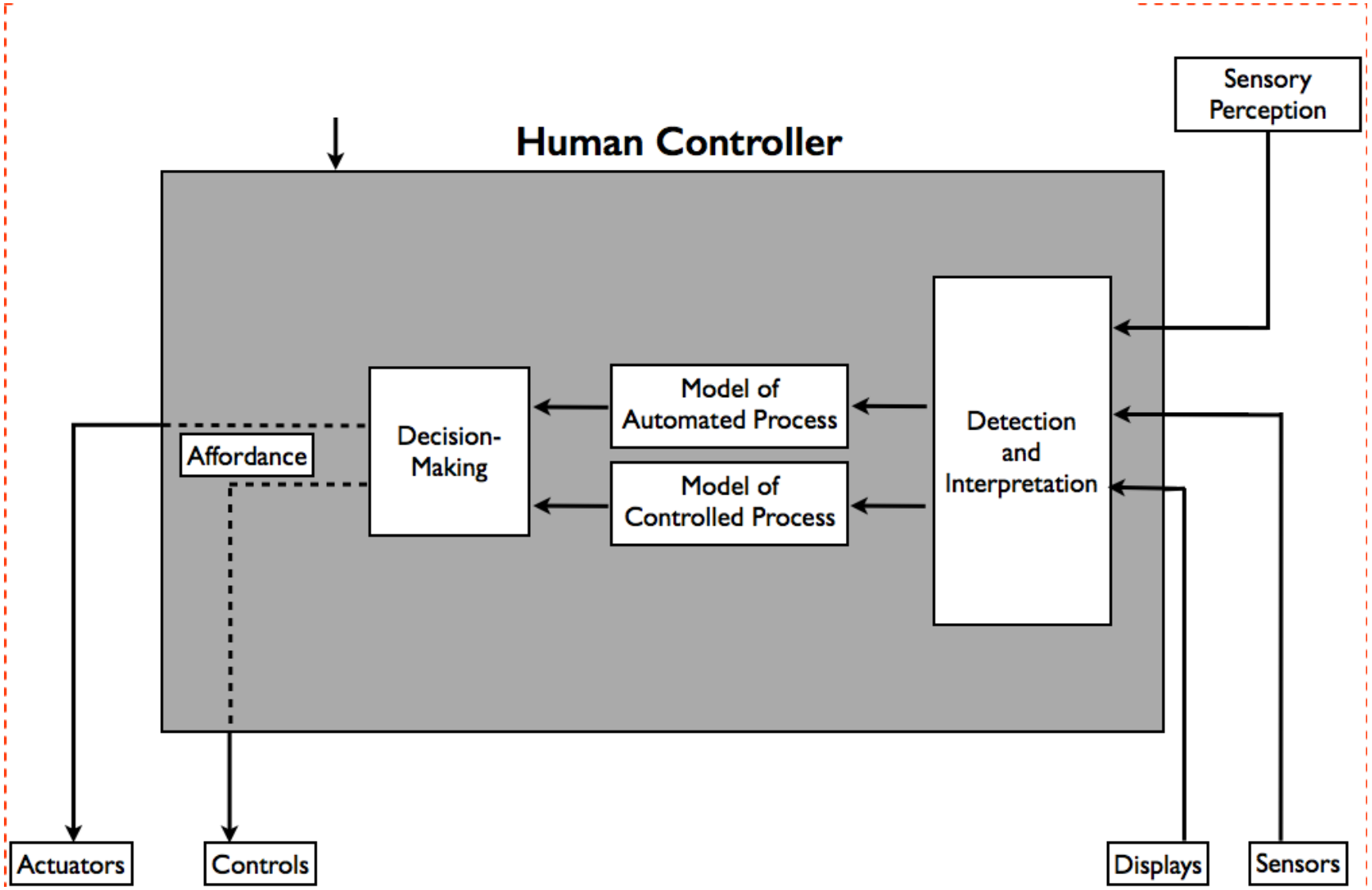


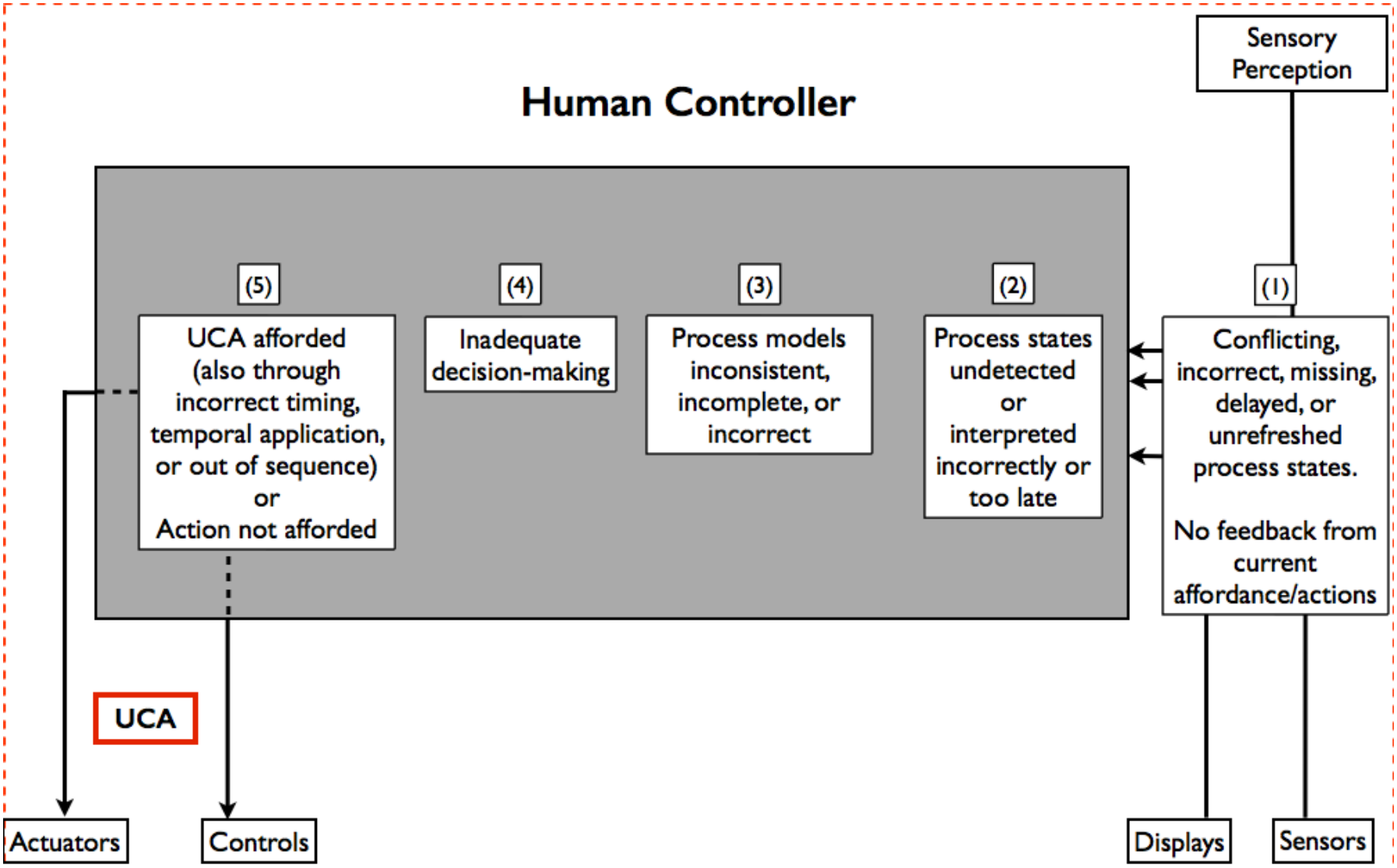
Example Fault Tree for Human Operator Behavior (adapted from RTCA, 2008)

STAMP Assumptions

- Human error is never a root cause
- Need to ask what led to that error in order to eliminate or reduce it
- The error almost always rooted in system design or in the context in which human working


Written/Trained Procedures
Environmental Inputs
Human Controller
Model of Context
Model of automation
Control Action Generation
Model of Controlled Process
Controls
Displays
Automated Controller
Environmental Inputs
Control Algorithm
Model of Controlled Process
Actuators
Sensors
Controlled Process
Process Inputs
Process Outputs
Disturbances


Human Controller
Sensory Perception
Detection and Interpretation
Model of Automated Process
Model of Controlled Process
Decision-Making
Affordance
Actuators
Controls
Displays
Sensors


Human Controller
Sensory Perception
(5)
UCA afforded (also through incorrect timing, temporal application, or out of sequence) or Action not afforded
(4)
Inadequate decision-making
(3)
Process models inconsistent, incomplete, or incorrect
(2)
Process states undetected or interpreted incorrectly or too late
(1)
Conflicting, incorrect, missing, delayed, or unrefreshed process states.
No feedback from current affordance/actions
UCA
Actuators
Controls
Displays
Sensors

Augmented Analysis

- Identify information controller needs and when needed (e.g., situation awareness)
- Identify detailed scenarios that could lead to the unsafe behavior (control actions), why human acted the way they did
- Use this information to improve the system design and reduce human errors

LEARN 1 Grant (1) Results

1. Developed new analysis technique (based on STAMP and systems theory) to be used on early concept analysis
   - Rigorous procedure to construct the models from the ConOps
   - Analysis procedures to analyze the model
2. STECA (System-Theoretic Early Concept Analysis) uses ConOps to identify
   1. Missing, inconsistent, conflicting safety-related information
   2. Vulnerabilities, risks, tradeoffs
   3. Safety requirements for rest of system life cycle
   4. Potential design or architectural solutions for hazard scenarios
   5. Information needed by humans and by automation to operate safely (process models)

LEARN 1 Grant (2)

3. Demonstrated STECA on TBO (Trajectory-Based Operations) ConOps

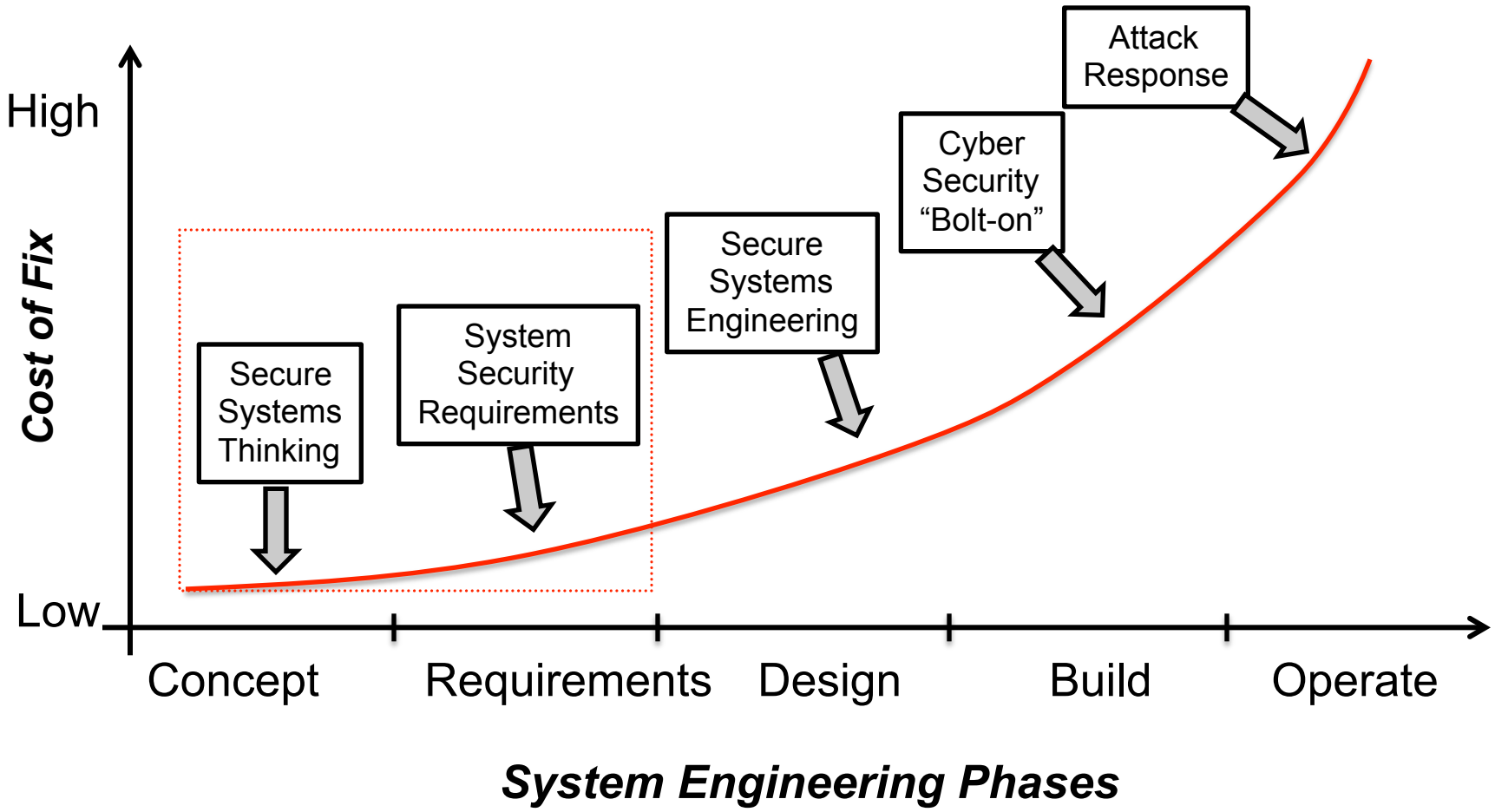
4. Compared it to results of TBO PHA (Preliminary Hazard Analysis)

5. Extended STAMP hazard analysis to include some sophisticated human factors concepts (e.g., situation awareness)

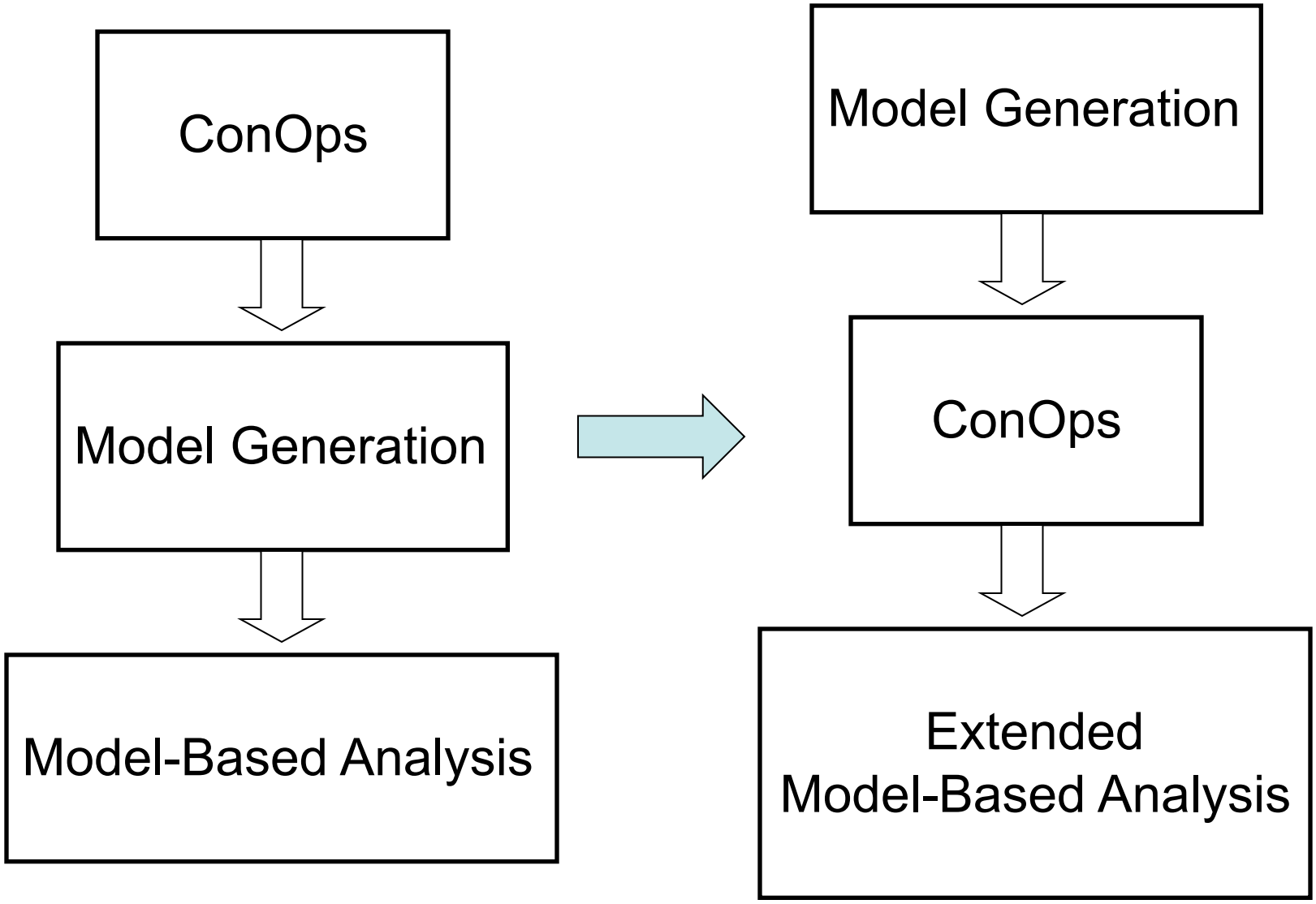
Potential LEARN 2 Research on Distributed Air Traffic Management

- Interested partners at: NASA Ames, NASA Langley, and JSC (Johnson Space Center)
- Topics:
  - Designing security into future air traffic management systems
  - Developing a formal ConOps development language.
  - Adding more human factors in the analysis (e.g., mode confusion)
  - Extending STECA and model-based analysis
  - UAV integration into NAS
  - Automated tools
  - Applying to most critical outstanding problems in distributed ATM

Build Security into ATC Like Safety

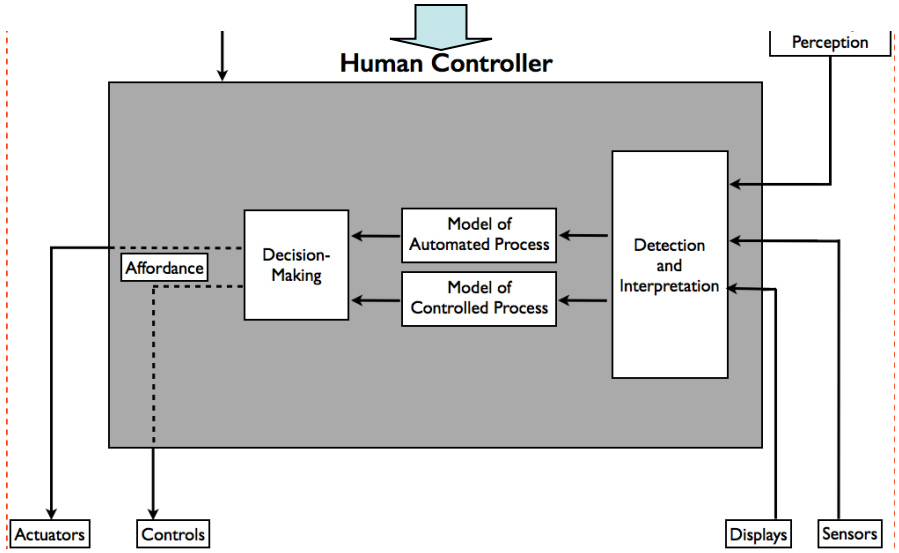


Model-Based System Engineering



Extend Human Aspects of Analysis

Written/Trained Procedures **Environmental Inputs** **Operational Culture** **Social Context** **Physiological Factors**



- Design to maintain situation awareness, avoid mode confusion, etc.
- What should lost link procedures be?
- How to trade between pilot/ATC and automation control authority
- Etc.

Extend General Analysis Capabilities

- Analysis of safety of centralized vs. distributed operations
  - Mismatches in information flow and control flow?
  - Mismatches in control flow and agent authority?
  - Missing/incorrect environmental assumptions/
  - Hazards related to collaborative decision making and action execution across a distributed system
- Analysis of modes and levels of uncertainty that can be tolerated
- Identifying agent-level assumptions necessary to limit system-wide uncertainty and assure global safety
- Modeling and analyzing timing requirements for safety
- Tradeoffs between different qualities: safety, stability, throughput, robustness
- Etc.

Apply to National Airspace System

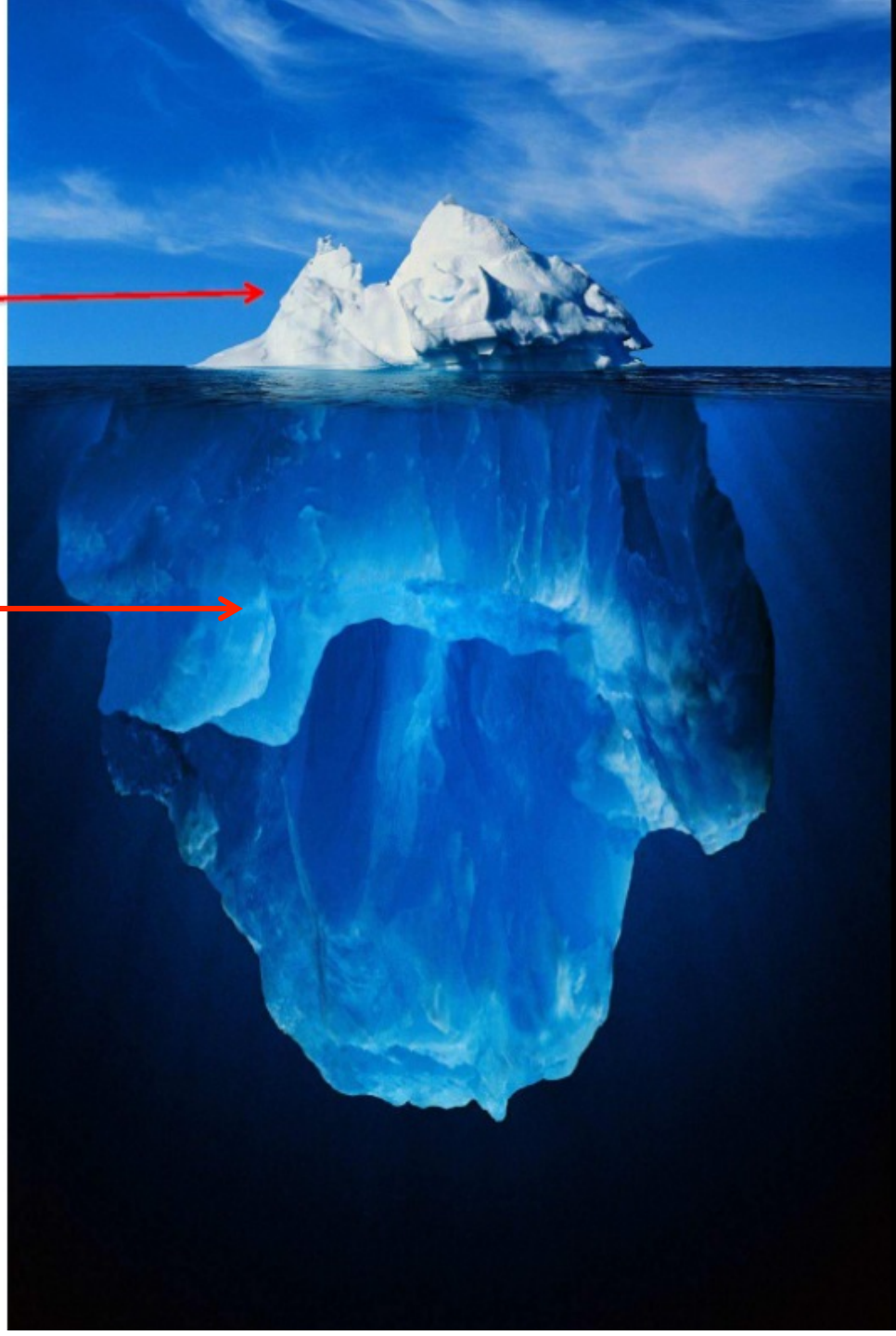
- Apply the new tools to most critical aspects of re-engineering the NAS
  - TBO versions and other proposed changes
  - Introduction of UAS into the NAS
    - Safety requires considering more than just DAA (Detect and Avoid)
    - What will impacts be on safety assumptions of current system? What changes will be needed?
    - For a mixed group of vehicles (manned, remotely piloted, unmanned), what control architectures will enable collaborative decision making that ensures safe separation?

LEVERAGE

Analytic Reduction

Systems Theory

(Allows seeing more of program space and evaluate potential Solutions)



WYOUNG@MIT.EDU